

The background of the slide features a light blue to green gradient. Overlaid on this are faint, large-scale binary digits (0s and 1s) and a series of thin, white, curved lines that form a dome-like structure, suggesting a network or data flow.

Primi passi Programmazione in C

Primo programma

```
#include <stdio.h>

int main()
{
    printf ("Hello World");
    return 0;
}
```

L'esecuzione di un programma inizia dal **main()**

Il corpo del **main** parte e finisce con una parentesi graffa **{ }**

istruzioni

```
#include <stdio.h>

int main()
{
    printf ("Hello World");
    return 0;
}
```

Nel **main** le istruzioni vengono eseguite in sequenza

Ogni istruzione finisce con un punto e virgola ;

commenti

```
#include <stdio.h>
/*primo programma in C*/
int main()
{
    printf ("Hello World");
    return 0;
} //fine programma
```

- I commenti iniziano con **/*** e terminano con ***/** su più righe
- Su una sola riga si può utilizzare **//**

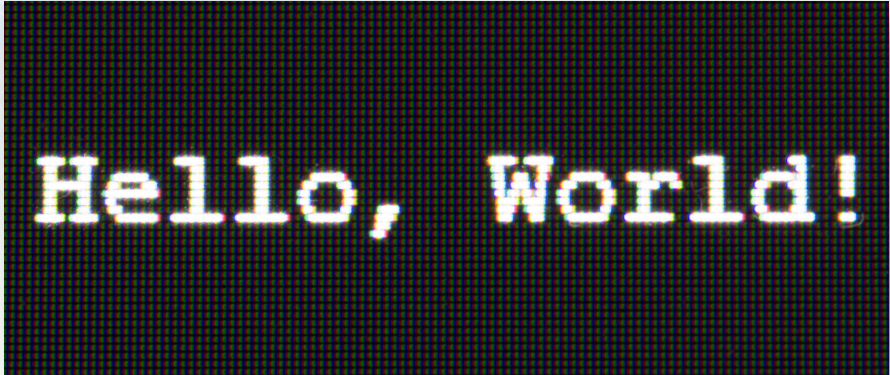
#include <stdio.h>

#INCLUDE è un'istruzione che
'include' nel programma le funzioni
dichiarate nel file **stdio.h** (*standard
input-output*)

Funzioni come **printf** che abbiamo
usato per scrivere "Hello World!" o
scanf che utilizzeremo fra poco...

printf

Printf stampa a video quello che è scritto tra doppi apici.



```
Hello, World!
```

Caratteri escape

Scrivendo una stringa con printf possiamo aver bisogno di alcuni caratteri speciali:

Carattere	Significato
\n	New line – a capo
\t	tabulazione
\\	Carattere \
\'	apostrofo

Variabili

Per leggere dei dati immessi in input dalla tastiera dall'utente abbiamo bisogno di dichiarare delle variabili

```
#include <stdio.h>

int main()
{ /* Inizio del main */
int x;
printf("Inserisci un numero:");
scanf("%d",&x);
printf("Il quadrato di %d e' %d\n",x,x*x);
return 0;
} /* Fine del main */
```


variabili

Con l'istruzione:

```
int x;
```

Dichiariamo una variabile **x** in cui l'utente andrà ad inserire un qualsiasi valore intero.

```
Inserisci un numero:55
Il quadrato di 55 e' 3025
```

scanf

La funzione scanf resta in attesa di un input da parte dell'utente e lo assegna alla variabile indicata.

```
scanf("%d",&x);
```

%d

Quando in una stringa, in **printf** o **scanf**, troviamo il simbolo **%** vuol dire che in quel posto andrà inserita una variabile di un certo tipo.

%d si aspetta un numero intero di tipo decimale.

if

Quando si desidera eseguire una istruzione solo se viene verificata una certa condizione, si utilizza if.

```
if(condizione)
```

```
{ ... }
```

```
else
```

```
{ ... }
```


if

```
#include <stdio.h>

int main()
{ /* Inizio del main */
int x;
printf("Inserisci un numero:");
scanf("%d",&x);
if(x%2 == 0)
    printf("Il numero %d e' pari\n",x);
else
    printf("Il numero %d e' dispari\n",x);
return 0;
} /* Fine del main */
```



ATTENZIONE

Non ho
inserito le
parentesi
graffe perché
è una sola
istruzione!

If risultato

A secondo dei diversi numeri inseriti dall'utente, cambia il risultato:

```
Inserisci un numero:33  
Il numero 33 e' dispari
```

```
Inserisci un numero:54  
Il numero 54 e' pari
```